

IN THE UNITED STATES PATENT AND TRADEMARK OFFICE
BEFORE THE BOARD OF PATENT APPEALS & INTERFERENCES

-----: Dkt: NETEX-P4-US (35817)
In Re U.S. Patent Application Of: : Examiner: Ke, Peng
REFUAH : Group No: 2174
Serial No.: 10/031,794 :
Filed: 01/17/2002 :
Title: USER INTERFACE METHOD -----:

APPELLANT'S BRIEF ON APPEAL

TO THE BOARD OF PATENT APPEALS & INTERFERENCES:

Appellant's Appeal Brief herein is submitted in furtherance of the Notice of Appeal filed in the U.S. Patent Office on December 6, 2010, in the above-captioned U.S. Patent Application.

Payment of the \$270 fee (SE) for filing the Appeal Brief is submitted herewith by authorization to charge Applicant attorney's Deposit Account.

Applicant appeals from the Final Rejection of Claims 1, 5, 6, 8-36, 38-63 and 79-88 by the Examiner in the Office Action dated August 5, 2010. The Final Rejection is believed to be in error and it is requested that it be reversed

on appeal.

A printed listing of Claims under Final Rejection is appended hereto in the Claims Appendix. Applicant's evidence relied on in the appeal is appended in the Evidence Appendix. The Appendix of Related Proceedings lists none.

REAL PARTY IN INTEREST

The real party in interest in the present Application on appeal is EasyNet Access Inc., New York, NY, which is the assignee of the entire interest in the present U.S. Patent Application.

RELATED APPEALS / INTERFERENCES

There are no other related applications, appeals or interferences to the present U.S. Patent Application.

STATUS OF CLAIMS

The total number of claims submitted in the application is: 88

Claims cancelled: 2-4, 7, 37, 64-78

Claims withdrawn from consideration but not cancelled: None

Claims pending: 1, 5, 6, 8-36, 38-63, 79-88

Claims allowed: None

Claims under final rejection and on appeal: 1, 5, 6, 8-36, 38-63, 79-88.

STATUS OF AMENDMENTS

Applicant filed a Response to the Final Rejection with Notice of Appeal on December 6, 2010, submitting a proposed amendment of terminology in the claims and further argument for patentability. By Advisory Action dated January 4, 2011, the Examiner refused entry of the proposed amendment to the claims stating that it would not place the application in condition for allowance. Applicant filed a Response to Advisory Action on January 5, 2011, maintaining the Notice of Appeal of the claims as rejected in the Final Rejection.

SUMMARY OF CLAIMED SUBJECT MATTER

The claimed subject matter is defined in main Claims 1, 6, 47, 56 and 85 of the present Application, of which main Claim 1 is representative of the claimed subject matter in all main claims, as follows:

Claim 1:

A method of executing a command entered in a URL field of a computer browser, which command is directed to achieve an action in a

separately-executing program other than said browser, comprising:

- (a) employing a WWW browser having a window with a designated URL field and a graphical display area of said browser;
- (b) receiving a text string, representing a command in a format which is neither a standard URL nor a portion thereof, which is entered into said designated URL field of said WWW browser;
- (c) translating, by computer, said command into at least one action to be executed by the other separately-executing program, wherein said text string comprises a multiplicity of words and said command is identified according to at least one of said multiplicity of words; and
- (d) enabling a display of a status of the other separately-executing program in the graphical display area of said browser while the user continues to employ said browser.

The above-indexed elements recited in main Claim 1 are described and supported in the Specification and the Drawings of the Application as follows:

- (a) Fig. 1 shows, and the Description on Page 7, Lines 23-34 describes, a WWW browser window 100 with a URL entry/display field 106 and a graphical display area 108. The invention enables a user to enter a command in the URL field 106 of the browser window 100 which interacts with other software running on the same computer or on a remote computer.
- (b) The Description on Page 8, Line 5, to Page 9, Line 19, describes that the command is entered in a format that is differentiated

from a standard URL, including: commands prefixed with a special code (Page 8, Lines 9-10); free-form natural language (Page 8, Lines 27-28); template language (Page 8, Lines 29-34); script commands, conditional statements, etc. (Page 9, Lines 1-2); or implicit or form-filling commands (Page 9, Lines 6-19).

(c) The Description on Page 11, Lines 7-22, describes that the command is translated into instructions for the other software packages (Line 11) or other appropriate instructions for other types of software (Lines 14-16). As noted in (c) above, the command may be composed of special codes, free-form natural language words, template words, script commands, conditional statements, implicit or form-filling words, etc. Fig. 2 and the Description on Page 16, Lines 6-29, also describes that the user interface system for command translation and execution is comprised of a client 202 with a browser 204 and command interface 206 for capturing user-input commands. Each captured command is parsed and translated into instructions (Line 14) to another separately executing program. The instructions are performed and the results are displayed at any of the client 202, remote processor 210, or remote computer 214.

(d) The Description on Page 9, Line 20, to Page 10, end, describes that the command entered in the URL field can affect the display area 108 of the browser window 100 in many ways, including: displaying command results in data display area 112 (Page 9, Lines 22-23); displaying what the system understood the user's command to mean or explaining the consequences of performing the command (Page 9, Lines 31-33); providing feedback on the results or progress or

other statuses of the command (Page 10, Lines 16-19); displaying the effect of the command (Page 10, Line 20). "The software controlled by the interface may be executed on a local or a remote computer" (Page 10, Lines 33-34).

Main Claim 1 does not contain any means-plus-function or step-plus-function recitation.

Main Claim 6 similarly recites the steps of employing a WWW browser of a presently executing computer program having a window with a designated URL field and a graphical display area, receiving a text string representing a command in a format that is not a standard URL in the URL field of the WWW browser (in a manner affecting translation of future commands), translating the command into action to be executed by the other separately-executing program, and displaying a status of the other separately-executing program in the graphical display area of the browser while the user continues to employ said browser.

Main Claim 47 similarly recites the steps of employing a presently executing program with a window (browser) having an input URL field and a graphical display area, entering a text string in the input URL field, parsing the text string to determine a command other than a domain (URL), executing the command to perform an action in the other separately-executing program,

and displaying a status of the other separately-executing program in the window of the presently executing program while the user continues to employ the presently executing program. The Description on Page 16, Line 30, to Page 17, end, also describes that the command interface may be a plug-in, a separate program, or interface other than a browser such as email software (Line 25) or a handwriting input field (Line 34).

Main Claim 56 similarly recites the steps of employing a browser with an interface that is separately executing from an executing program, receiving input of a command in the browser interface directed to the separately-executing program, causing a response to the command to be returned in the browser interface, and displaying the response to the command to the browser interface while the user continues to employ the browser interface.

Main Claim 85 adapted for making a telephone connection similarly recites the steps of employing a WWW browser having an input URL field and a graphical display area, entering a text string representing a command that is not a standard URL in the input URL field, translating the command to perform an action in the other separately-executing program, and displaying feedback from the separately-executing program in the graphical display area of the browser while the user continues to employ the browser. The Description on Page 12, Lines 11-21, also describes using the interface

method to make a telephone connection or other messaging contact.

The other depending Claims 5, 8-36, 38-46, 54-55, and 79-83 dependent on Claim 1, Claims 48-53, 84, and 86 dependent on Claim 47, Claims 57-63 and 87-88 dependent on Claim 56 define ancillary features of the system, and are deemed to stand or fall on the same point of patentability as presented for the main claims on which they depend.

GROUNDS OF REJECTION TO BE REVIEWED ON APPEAL

In the final Office Action dated August 5, 2010, the Examiner rejected all claims under 35 U.S.C. 112, first paragraph as failing to comply with the written description requirement. The Examiner stated that the Specification did not provide support for the limitation "enabling a display of a status of the other separately-executing program in the graphical display area of said browser while the user continues to employ said browser". The Examiner maintained that the Specification only provides support for "displaying the status of the executed command not the status of the application".

In the final Office Action, the Examiner further rejected all claims under 35 U.S.C. 103(a) as being unpatentable over the Estabrook reference, Teach Yourself Microsoft Internet Explorer 4 in 24 Hours ("Estabrook"), in view of the W3C reference, Line Browser Commands ("W3C").

ARGUMENT

As to the rejection of the claims under 35 U.S.C. 112, the Examiner further explained in comments in the Advisory Action that the claim recitation of "enabling a display of a status of the other separately-executing program in the graphical display area of said browser while the user continues to employ said browser" means that the executing program is not a separate program from the browser. The Examiner stated that in the examples Applicant pointed to as displaying a status or effect of the other separately-executing program, "both instances show a HTML page that is displayed within the browser".

It is submitted that the Examiner's interpretation of the phrase "displaying a status or effect of the other separately-executing program in the browser" as being a display of an HTML page that is not separate from the browser is clearly erroneous and a complete misreading of the Applicant's Specification. The Specification's entire description explains Applicant's invention of using the URL field of a browser to input a command to control an action in a separately-executing program from the browser, and then displaying the status or effect of that action in the graphical display area of the browser. The Specification is replete with examples of using the browser

URL field to enter a command and the browser graphical display area to display the status or effect of the command for a wide range of other software types, packages, and locations executing separately from the browser. All claims in their finally rejected form clearly recite this novel function. The Examiner's disregarding of the description of many examples of separately-executing programs in the Specification, and instead reading the claim recitation of the display of the status or effect of the command in the browser as itself an instance of a browser display that is not a separate program from the browser, is an inapt reading of the totality of Applicant's Application and should be reversed.

As to the rejection of the claims under 35 U.S.C. 103(a), the Examiner maintained that the Estabrook reference, page 132, Figs. 9.4 and 9.5, discloses enabling a display of an effect of a separately-executing program in the display of the browser. However, the separately-executing program the Examiner cited in Estabrook is the History Frame function of the browser itself. It is well known in the relevant industry that the History Frame used in a browser is a browser function, not a separately-executing program. Therefore, it appears that the Examiner's interpretation of the History Frame of the browser as a separately-executing program in the Estabrook prior art is incorrect and improperly applied against the key limitation in the claims of

"enabling a display of a status (or effect, response, result, or feedback) of the other separately-executing program in the graphical display area of said browser while the user continues to employ said browser".

Nor can the Examiner maintain a position that the claimed "enabling a display of a status of the other separately-executing program in the graphical display area of said browser while the user continues to employ said browser" is obvious over the Estabrook reference teaching only use of the browser's History Frame function. It is well known in the relevant industry that the History Frame function in a browser is a browser function, not a separately-executing program. Therefore, the position that controlling and displaying the status or effect of a separately-executing program from within the browser is obvious over a reference that only teaches the History Frame function of the browser itself would be an entirely undocumented ground of rejection.

The USPTO MPEP states, "To support the conclusion that the claimed invention is directed to obvious subject matter, either the references must expressly or impliedly suggest the claimed invention or the examiner must present a convincing line of reasoning as to why the artisan would have found the claimed invention to have been obvious in light of the teachings of the references." See "Contents of a 35 U.S.C. Rejection", MPEP 706.02(j).

The Examiner has not presented a convincing line of reasoning as to

why the ordinary artisan in the field would have found that controlling and displaying the status or effect of a separately-executing program from within the browser would be obvious over a reference that only teaches the History Frame function of the browser itself. No evidence exists in the record of any prior suggestion of the invention as claimed. The only suggestion to do this comes from the Applicant's own disclosure. Therefore, the Examiner's holding of obviousness amounts to hindsight reconstruction of the Applicant's invention based only on the benefit of the Applicant's own disclosure.

CONCLUSION

In summary, the Examiner's final rejection of Claims 1, 5, 6, 8-36, 38-63, 79-88 under 35 U.S.C. 112 as unsupported by the Specification and under 35 U.S.C. 103(a) as unpatentable over the cited prior art is in error and should be reversed.

Respectfully submitted,
ATTORNEYS FOR APPLICANT



Leighton K. Chong, USPTO Reg. 27,621
Michael Doyle, USPTO Reg. 29,635
Joshua Hamburger, USPTO Reg. 59,955
Ostrager Chong Flaherty & Broitman, P.C.
570 Lexington Avenue, 17th Floor
New York, NY 10022-6894
Tel: (212) 681-0600, Fax: (212) 681-0300

CLAIMS APPENDIX

1. A method of executing a command entered in a URL field of a computer browser, which command is directed to achieve an action in a separately-executing program other than said browser, comprising:
 - employing a WWW browser having a window with a designated URL field and a graphical display area of said browser;
 - receiving a text string, representing a command in a format which is neither a standard URL nor a portion thereof, which is entered into said designated URL field of said WWW browser;
 - translating, by computer, said command into at least one action to be executed by the other separately-executing program, wherein said text string comprises a multiplicity of words and said command is identified according to at least one of said multiplicity of words; and
 - enabling a display of a status of the other separately-executing program in the graphical display area of said browser while the user continues to employ said browser.

(Claims 2-4, canceled)

5. A method according to claim 1, wherein said step of receiving a text string also includes receiving a command is directed to modifying a page-presenting behavior of said WWW browser from a first behavior to a second behavior, such that presentation of a WWW page under the second behavior would be different than presentation of the WWW page under said first

behavior.

6. A method of executing a command entered in a text input area of a computer program which is presently executing on a computer, which command is directed to achieve an action in a separately-executing program other than the presently executing computer program, comprising:

employing a WWW browser of the presently executing computer program having a window with a designated URL field and a graphical display area of said browser;

receiving a text string, representing a command in a format which is neither a standard URL nor a portion thereof, which is entered into said designated URL field of said WWW browser;

translating, by computer, said command into at least one action to be executed by the other separately-executing program, wherein said step of receiving a text string also includes receiving a command is directed to affecting a translation of a future command entered into said designated URL field into an action of said other separately-executing program; and

enabling a display of a status of the other separately-executing program in the graphical display area of said browser while the user continues to employ said browser.

(Claim 7, canceled)

8. A method according to claim 1, wherein said action has a physical manifestation outside of any computer hardware of the computer on which the presently-executing program is executing.

9. A method according to claim 8, wherein said manifestation comprises making a telephone connection.
10. A method according to claim 8, wherein said manifestation comprises printing a file.
11. A method according to claim 1, wherein said action is performed on a same computer as is executing said browser.
12. A method according to claim 1, wherein said action is performed on a computer remote from a computer executing said browser.
13. A method according to claim 1, wherein said command is translated on a same computer as is executing said browser.
14. A method according to claim 1, wherein said command is translated on a computer remote from a computer executing said browser.
15. A method according to claim 1, comprising parsing said text to yield said command.
16. A method according to claim 15, wherein said parsing is performed on a same computer as is executing said browser.
17. A method according to claim 15, wherein said parsing is

performed on a computer remote from a computer as executes said browser.

18. A method according to claim 1, wherein said action is affected by a context for translating said command.

19. A method according to claim 18, wherein said context affects said translation.

20. A method according to claim 18, wherein said context affects a parsing of said text into said command

21. A method according to claim 18, wherein said context affects one or more parameters associated with said command.

22. A method according to claim 18, wherein said context comprises a virtual personality of a user using said browser.

23. A method according to claim 18, wherein said context comprises a WWW page displayed by said browser.

24. A method according to claim 18, wherein said context comprises a state of at least one software package other than said browser.

25. A method according to claim 24, wherein said software package is executing on a same computer as said browser.

26. A method according to claim 18, wherein said context comprises a current state of affairs.
27. A method according to claim 18, wherein said context comprises a history of a state of affairs.
28. A method according to claim 27, wherein said history comprises a history of actions by a machine.
29. A method according to claim 27, wherein said history comprises a history of data display.
30. A method according to claim 27, wherein said history comprises a history of user input.
31. A method according to claim 1, wherein said command has an affect on future actions dictated by future commands.
32. A method according to claim 1, wherein said text string contains said command in an explicit manner.
33. A method according to claim 1, wherein said text string contains said command in an implicit manner.
34. A method according to claim 33, wherein said command is determined responsive to an identification of a type of data comprised in the

text string.

35. A method according to claim 1, wherein said command comprises a natural language format command.

36. A method according to claim 1, wherein said command comprises a fixed format command.

(Claim 37, canceled)

38. A method according to claim 1, wherein said graphical display area of said browser displays a result of said action.

39. A method according to claim 1, wherein said graphical display area of said browser displays a status report on said action.

40. A method according to claim 1, wherein said graphical display area of said browser is displayed asynchronously.

41. A method according to claim 1, wherein said graphical display area of said browser is generated on a same computer as is executing said browser.

42. A method according to claim 1, wherein said graphical display area of said browser is generated on a computer remote from a computer executing said browser.

43. A method according to claim 1, wherein said graphical display area of said browser comprises a request to clarify said action.

44. A method according to claim 1, wherein said graphical display area of said browser is modified in real time responsive to said command.

45. A method according to claim 44, wherein said display comprises a multi-media stream.

46. A method according to claim 1, wherein said graphical display area of said browser modifies a previously displayed data page on said browser.

47. A method of performing an action in a separately-executing program other than a presently executing computer program on a computer, comprising:

employing the presently executing computer program having a window with an input URL field and a graphical display area;

receiving a text string which is entered in the input URL field which is executing with the presently executing computer program on the computer;

parsing said string to determine a command to achieve said action, said parsing being at a location other than a domain indicated by said entered text string;

executing said command to perform said action in the separately-executing program other than the presently executing computer program; and

enabling a display of a status of the other separately-executing program in the graphical display area of said browser while the user continues to employ said presently executing computer program.

48. A method according to claim 47, wherein said string is a standard URL.

49. A method according to claim 47, wherein said string is not a standard URL.

50. A method according to claim 47, wherein providing said text string comprises entering said string in an input field for a URL of a browser program executing on the computer.

51. A method according to claim 47, wherein providing said text string comprises providing said string in parameter position reserved for a URL in a network programming language.

52. A method according to claim 51, wherein said language comprises Java.

53. A method according to claim 51, wherein said language comprises HTML.

54. A method according to claim 1, wherein said browser displays live information from the Internet.

55. A method according to claim 12, wherein said remote computer communicates with said browser over the Internet.

56. A method of interacting with an executing program having one or more interfaces running on a computer, comprising:

employing a browser which is executing separately with the executing program, said browser including an interface, which is not one of the executing program's one or more interfaces;

receiving an input of a command in the interface of said browser directed to the executing program which is to provide a response to be returned to said browser interface;

causing a response to said command directed to the executing program to be returned to said browser interface; and

enabling a display of a status of the other separately-executing program in the browser interface while the user continues to employ said browser interface.

57. A method according to claim 56, wherein said response is displayed by said browser.

58. A method according to claim 56, wherein said separately executing program comprises a program executing on a same machine as said browser.

59. A method according to claim 56, wherein said separately

executing program comprises a program executing on a machine remote from a machine executing said browser.

60. A method according to claim 59, wherein said two machines are connected via the Internet.

62. A method according to claim 56, wherein said command is entered in a URL field of said browser.

63. A method according to claim 56, wherein said command is entered by interacting with a graphical display on said browser.

(Claims 64-78, canceled)

79. A method according to claim 1, comprising:
displaying a page having a plurality of options for making said action;
and
selecting, by the user, one of said plurality of options,
wherein making said action comprises using said one of said plurality of options.

80. A method according to claim 1, wherein said action comprising sending an SMS to a telephone.

81. A method according to claim 1, wherein said action comprising making a telephone connection to a telephone.

82. A method according to claim 79, wherein said action comprising sending an SMS to a telephone.
83. A method according to claim 1, wherein said action comprises at least one of copying data and modifying files.
84. A method according to claim 47, wherein said action comprises at least one of copying data and modifying files.
85. A method of making a telephone connection through a separately-executing program other than a presently executing program on a computer, comprising:
- employing a WWW browser of the presently executing computer program having a window with a designated URL field and a graphical display area;
 - receiving a text string, representing a command in a format which is neither a standard URL nor a portion thereof, which is entered into said designated URL field of said WWW browser; and
 - translating, by machine, said command into at least one action comprising making a telephone connection through the separately-executing program; and
 - causing a display, on said graphical display area of said WWW bowser, of feedback from said other separately-executing program while the user continues to employ the WWW browser.

86. A method according to claim 47, wherein said text string comprises a multiplicity of words.
87. A method according to claim 56, wherein the input to the browser interface comprises personal contact information associated with a person, and the other executing program is a communication program which executes communication with the person associated with the personal contact information entered in the browser interface.
88. A method according to claim 56, wherein the input to the browser interface is a product or service name associated with a website for purchasing of the product or service, and the other executing program is a purchasing program of the website for purchasing the product or service, and wherein confirmation of execution of the purchase is returned to the browser interface.

EVIDENCE APPENDIX

1. "Contents of a 35 U.S.C. §103 Rejection", MPEP 706.02(j)

APPENDIX: RELATED PROCEEDINGS

None.